Long short-term memory networks with attention learning for high-rate structural health monitoring

Matthew Nelson[a], Vahid Barzegar[b], Simon Laflamme[c], Chao Hu[d], and Jacob Dodson[e]

[a,b,c]Department of Civil, Construction, and Environmental Engineering, Iowa State University, Ames, IA, USA
[d]Department of Mechanical Engineering, Iowa State University, Ames, IA, USA
[e]Air Force Research Laboratory, Munitions Directorate, Fuzes Branch, Eglin Air Force Base, FL, USA

ABSTRACT

High-rate dynamic systems undergo events of amplitudes greater than 100 g_s in a span of less than 100 ms. The unique characteristics of high-rate dynamic systems include 1) large uncertainties in the external loads, 2) high levels of non-stationarity and heavy disturbances, and 3) unmolded dynamics generated from changes in the system configurations. This paper presents a deep learning algorithm consisting of an ensemble of long short-term memory (LSTM) cells used to conduct high-rate state estimation. The ensemble of LSTMs receives and transforms the signal into inputs of different time resolutions. Each input vector correlates to an LSTM cell which predicts the signal in real-time and produces feature vectors. The feature vectors are then processed through an attention layer and dense layer to predict the physical features of the system. Here, we study the temporal evolution of the attention layer weights to conduct state estimation, while the LSTM cells are attempting to conduct measurement predictions. We study the performance of the algorithm on experimental data generated by DROPBEAR, a dedicated testbed for high-rate structural health monitoring research. State estimation consists of estimating, in real-time, the location of a cart that moves along a beam. Results show that the attention layer weights can be used to estimate the cart location but that the beam requires impact excitations to accelerate the convergence of the algorithm.

Keywords: High-rate, Long short-term memory, Times series, Recurrent neural network, Non-stationary, Real-time, State estimation

1. INTRODUCTION

High-rate dynamic systems are defined as engineering systems experiencing accelerations of high amplitudes, typically higher than 100 g_n, over short durations, often less than 100 ms. Examples of such systems include blast mitigation mechanisms, advanced weaponry, and hypersonic vehicles. The field deployment and safe operation of high-rate systems require feedback capabilities in the sub-millisecond range and thus high-rate state estimation capabilities, here termed high-rate structural health monitoring (HRSHM).[1]

However, the development of HRSHM algorithms is a complex task given the unique characteristics of their dynamics that combine 1) large uncertainties in the external loads, 2) high levels of non-stationarity and heavy disturbances, and 3) unmolded dynamics generated from changes in the system configurations.[2] There have been notable efforts in literature in developing algorithms enabling HRSHM. For instance, Dodson et al.[3] proposed HRSHM using modal decomposition. Joyce et al.[4] formulated a sliding mode observer-based algorithm to estimate the position of a dynamic cart moving along a cantilever beam by estimating the system's natural

Further author information: (Send correspondence to M.N.)
M.N.: e-mail: mjn5@iastate.edu
VB.: e-mail: barzegar@iastate.edu
S.L.: e-mail: laflamme@iastate.edu
C.H.: e-mail: chaohu@iastate.edu
J.D.: email: jacob.dodson.2@us.af.mil

frequency in the time domain. The experimental setup used by the authors, termed DROPBEAR (Dynamic Reproduction Of Projectiles in Ballistic Environments for Advanced Research), was also used by others to validated HRSHM algorithms. In particular, Downey et al.[5] estimated the position of the cart through real-time model matching using the system's fundamental frequency extracted through a Fourier transform. Yan et al.[6] developed a sliding mode observer to track a reduced-order physical representation of the dynamics and numerically demonstrated HRSHM applicability.

While these physics-driven techniques showed great promise at HRSHM, they were developed and demonstrated on dynamics that were typically simpler than those experienced by real-world high-rate systems. Other works focused on high-rate time series predictions, where data-based techniques showed better applicability. For instance, Hong et al.[1] developed a real-time on-the-edge neural network to predict highly non-stationary data produced by accelerated Drop tower tests. A particularity of the technique is that it leveraged an input space that varied depending on the time series' local dynamics. An improvement of the algorithm was proposed in Barzegar et al.[7] that consisted of an ensemble of recurrent neural networks (RNNs) with long short-term memory (LSTM) cells arranged in parallel, where each RNN was constructed with a different input space in order to provide multi-time resolution capabilities. The algorithm showed great predictive capabilities with an average computation time of 25 μs.

In this paper, we extend our work on the ensemble of LSTM cells and evaluate its capability to perform state estimation, instead of solely time series prediction. The problem of state estimation is of great interest in the field of structural health monitoring because these estimates usually provide or map to actionable insights on the current/future states of structural systems.,[8,9] especially insights for state estimation.,[10,11] Here, the algorithm from Barzegar et al.[7] is applied to DROPBEAR data to study whether this algorithm can be utilized to quickly determine the position of the dynamic cart. This is done by pre-training each LSTM using the dynamics of the system under a given fixed position, assembling the pre-trained RNNs through an attention layer, and evaluating the evolution of the attention layer's weights from the real-time learning process during a real-time event.

The rest of the paper is organized as follows. Section 2 explains the architecture of the ensemble of LSTM cells. Section 3 describes the DROPBEAR testbed and the algorithm setup. Section 4 discusses the experimental results. The paper is concluded in Section 5.

2. ENSEMBLE OF LSTMS ARCHITECTURE

The time series prediction model is a deep learning model built to generate predictions sequentially, developed primarily to learn non-stationary time series on-the-edge. The model architecture is illustrated in Fig. 1. It consists of an ensemble of LSTM cells arranged in parallel, where each LSTM samples a different delay vector $\mathbf{x}_k^i$ at discrete time k to form its input space. The ith delay vector consists of observations sampled with a time delay τ_i and embedded in a vector of dimension d_i, with:

$$\mathbf{x}_k^i = \begin{bmatrix} x_{k+1-d_i\tau_i} & x_{k+1-(d_i-1)\tau_i} & \cdots & x_{k+1-2\tau_i} & x_{k+1-\tau_i} \end{bmatrix} \tag{1}$$

where τ and d are positive integers. Figure 1(left) shows a time series being sampled at different rates to generate each $\mathbf{x}_k$. It follows that the LSTMs extract features $\mathbf{h}$ of varying time resolutions. An attention layer assembles the LSTM outputs by linearly scaling the LSTM features $\mathbf{h}$ based on their relevance with the prediction target ($\hat{x}_{k+1}$). After, a dense layer takes the concatenated vector from the output of the attention layer and condenses features yielding $\hat{x}_{k+1}$ using a linear operation on the extracted features. The weights of each LSTM cell are updated using backpropagation by minimizing the following loss function.

$$\Gamma = \frac{\left(x_{k+1} - \hat{x}_{k+1}\right)^2}{2} \tag{2}$$

where x_{i+1} is the measured value, and $\hat{x}_{i+1}$ is the predicted value. The feature extractor (i.e., the LSTMs) weights are kept fixed and are trained using limited training data, as explained in what follows.

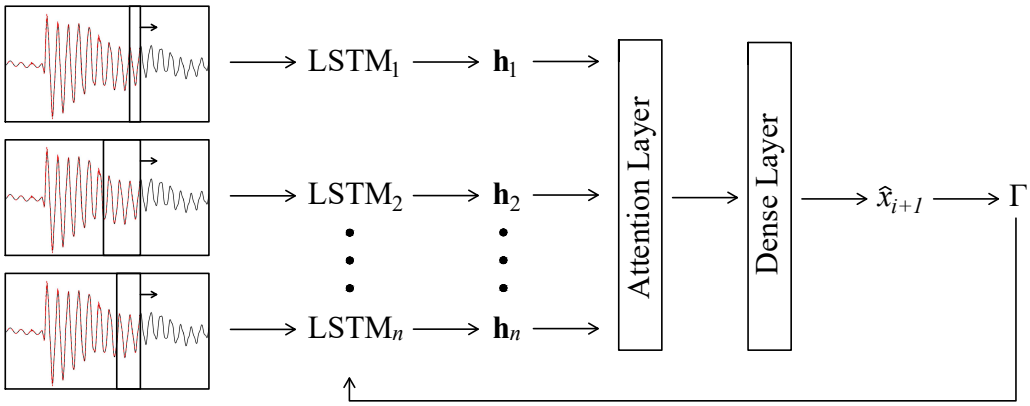


Figure 1: Architecture of the ensemble of LSTMs.

3. METHODOLOGY

The capability of the algorithm at conducting state estimation was verified on data collected from the DROPBEAR testbed. DROPBEAR, shown in Fig. 2, consists of a steel cantilever beam equipped with a dynamic cart moving along the beam and a mass attached with an electromagnet to reproduce sudden and rapidly varying changes in boundary conditions. An impact hammer is used to excite the beam. In this paper, we only consider the movement of the cart, which has rollers on top of and below the beam that act as a dynamic pin support.

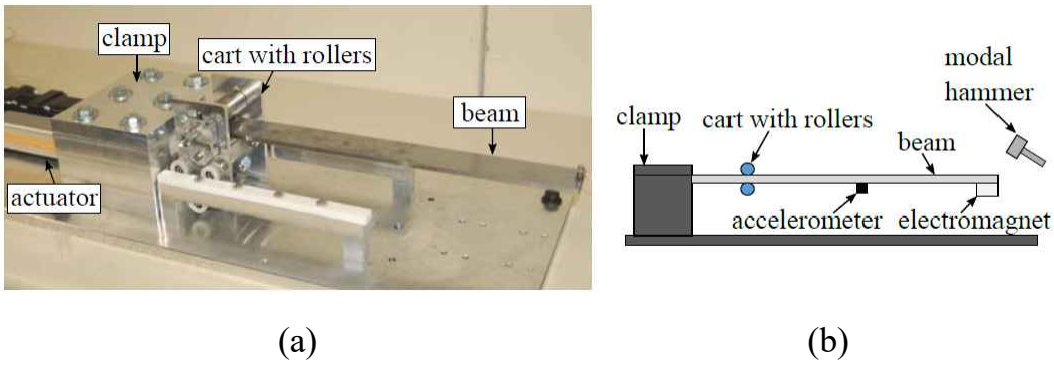


Figure 2: DROPBEAR Testbed.[6]

An accelerometer (PCB 353B17) was placed 400 mm away from the clamp. Acceleration measurements were recorded at a sampling rate of 25 kHz. Tests were conducted under fixed and dynamic conditions. For the fixed cart tests, the cart was placed 50 mm, 100 mm, 150 mm, and 200 mm away from the clamp. Five tests per position were performed and the responses measured following impacts at the tip of the beam. For the dynamic cart tests, the cart was moved from 50 mm to 250 mm over a period of 100 ms, and back to 50 mm over another period of 100 ms. Two different settings were used when collecting acceleration time series. In the first setting, the dynamic cart was the only vibration source, while in the second setting the beam was impacted with the modal hammer four times during the test. Figure 3(a) plots a typical acceleration time series for the fixed cart at 50 mm, and Fig. 3(b) plots the acceleration time series during the dynamic cart without impact excitations.

The LSTM cells were constructed and trained on simple physical knowledge, as explained in the upcoming subsection. Performance of the ensemble model at detecting the cart position through the attention layer weights

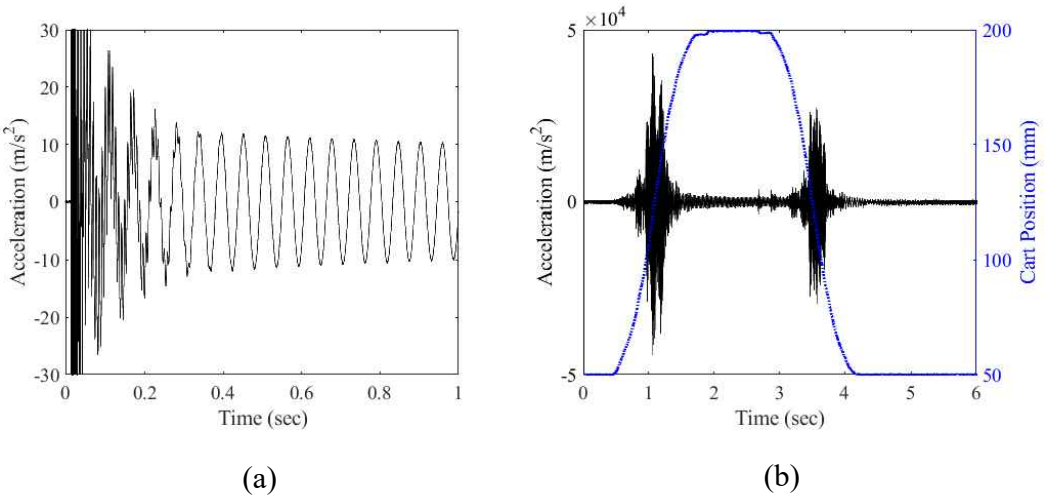


Figure 3: Typical acceleration time-series responses obtained from the DROPBEAR tests for the (a) static cart positioned 50 mm away from the clamp and (b) dynamic cart without hammer impacts.

was evaluated under both the fixed and dynamic tests. Convergence time was used as a performance metric. It is defined as the time required for the network to predict the next step under an estimation error threshold of 10%. The fixed cart tests were used to verify the capability of the LSTM ensemble at detecting the cart locations under vibration signatures that resemble those used for training. The dynamic cart tests were used to evaluate the performance of the algorithm at detecting the cart position under more complex dynamics.

3.1 Training of the LSTMs

For the construction and training of the LSTM network, we used knowledge of the dynamics for the four fixed cart positions. Thus, four LSTMs were used in the ensemble, each trained under the dynamics of a given fixed position. For each LSTM, $\mathbf{x}$ was selected to best represent the dynamics of a damped harmonic of frequency equal to that of the beam's fundamental frequency under the given cart position. As schematized in Fig. 4, this was done under 17.8 Hz, 21 Hz, 25 Hz, and 31 Hz vibrations, which covered the range of frequencies experienced by the dynamic cart. Under each frequency, the time delay τ was selected based on the mutual information (MI) method,[12] after which the embedding dimension d was selected heuristically to yield the best prediction performance out of each LSTM cell.

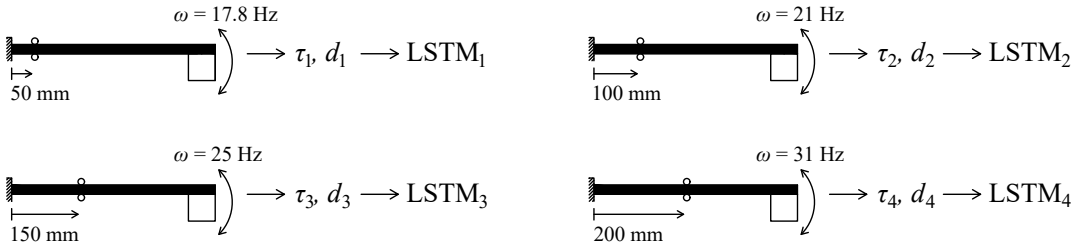


Figure 4: Input space selection for each LSTM cell.

Following the input space selection, each LSTM cell was individually trained on damped free vibration time series of frequency equal to that of the corresponding cart position. The LSTM cells were implemented in Python 3.7 using the Keras package with the mean squared error loss function and hyper-parameters listed in Table 1.

The weights of the LSTM cells were then transferred to the ensemble architecture to predict the dynamics of DROPBEAR in real-time.

Table 1: Hyperparameters used for training each LSTM cell.

LSTM	Learning rate	Hidden units	τ	d
1	0.015	50	367	3
2	0.015	50	315	7
3	0.010	50	261	6
4	0.008	50	215	6

4. NUMERICAL RESULTS

This section reports the numerical results. Results from the fixed cart tests are first presented, followed by those from the dynamic cart tests.

4.1 Fixed Cart Tests

Figure 5 plots the results taken from a representative test under each cart location. It can be observed from the acceleration time series (top) and estimation errors (middle) that the ensemble of LSTMs successfully converges under each case, with the convergence time increasing with the increasing system frequency (i.e., cart position). Results also show the evolution of the attention layer weights (bottom). Under each fixed location, a single weight converges at a significantly higher value compared to the other weights post-convergence. Importantly, the highest weight multiplies the output of the LSTM that was trained with the correct cart location (LSTM 1 for the cart at 50 mm, LSTM 2 for the cart at 100 mm, LSTM 3 for the cart at 150 mm, and LSTM 4 for the cart at 200 mm). These results suggest that the attention layer weights can be used to determine the fixed cart locations, where the dominating weight can be linked to the actual position.

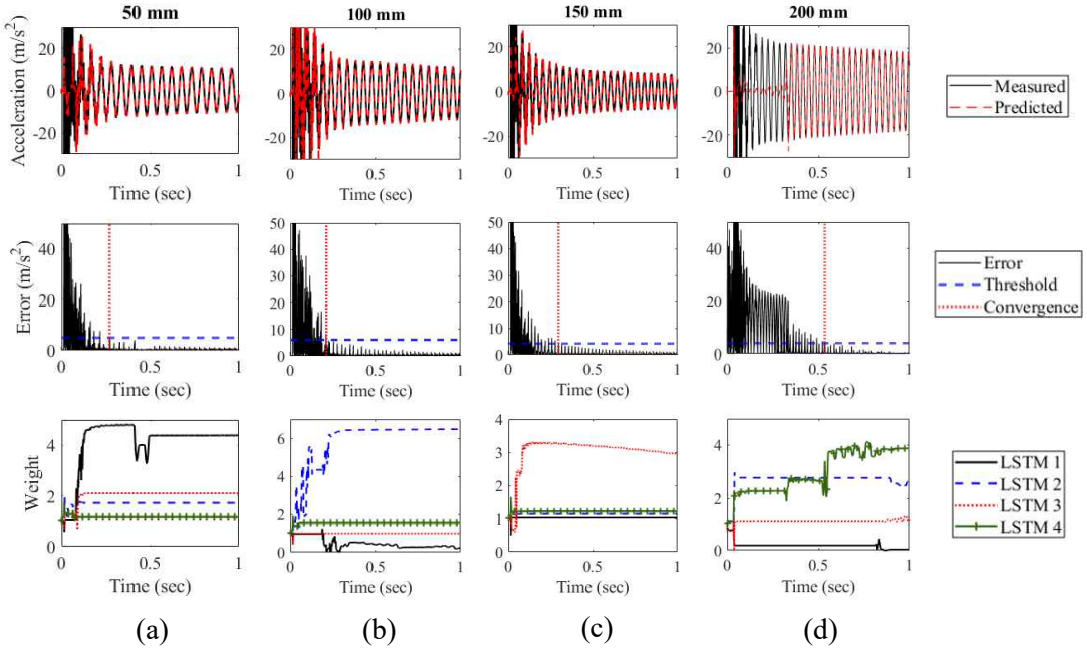


Figure 5: Typical results under each fixed cart location, showing the acceleration time series (top), estimation error (middle), and evolution of attention layer weights (bottom).

Table 2 lists the average convergence times obtained under the four fixed cart positions. The results are benchmarked against those reported in,[6] where the authors developed a model-reference adaptive system (MRAS)-based real-time algorithm to detect the position of the cart. Results show a convergence time in the hundreds of milliseconds instead of a desired sub-millisecond realm, attributable to the imposed null initial conditions. It is hypothesized that less aggressive initial discrepancies would lead to faster convergence; this is left to future work. It can also be observed that the convergence time achieved by the ensemble of LSTMs increases with the increasing cart position, and this trend is opposite to that from the results by the MRAS-based algorithm. This phenomenon may be attributable to the learning rates that provoked a strong initial overshoot of the stiffness estimated by the MRAS-based algorithm at a low frequency, and the initial overshoot diminished with the increasing frequency.

Table 2: Average convergence time - fixed cart tests.

Tests	Fundamental frequency (Hz)	Cart position (mm)	Ensemble of LSTMs (ms)	MRAS[6] (ms)
1-5	17.8	50	240	780
5-10	21.0	100	243	400
11-15	25.0	150	300	160
16-20	31.0	200	441	100

4.2 Dynamic Cart Tests

Results from the dynamic cart tests are plotted in Fig. 6, showing the acceleration, error, and attention weights time series without impact excitations (Fig. 6(a)) and with impact excitations (Fig. 6(b)). The times of the four impacts can be observed in the acceleration time series (Fig. 6(b), top) occurring at 390, 2170, 4160, 6240 ms. A study of the error plots reveals that convergence of the algorithm is faster with impacts, which can be attributed to the richer excitations.

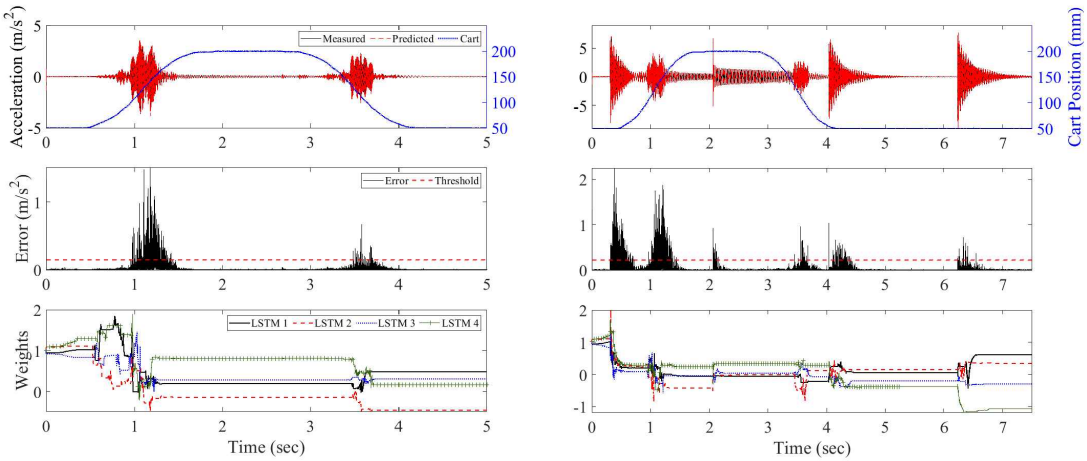
Before testing, a Butterworth filter was applied to the data to reduce noise. The Butterworth filter is a high and low pass filter that flattens the frequency response of the data. This filter keeps the needed features while limiting the noise from testing. The Butterworth filter used was a first-order high pass with cutoff frequencies of 10 Hz and 400 Hz, implemented using MATLAB built-in functions. The data was scaled down by a factor of 10^{-4} so the testing data would be similar to the training data.

Table 3 compares the convergence time of the ensemble of LSTMs to those reported in[6] using the MRAS-based real-time algorithm. Remark that the authors in[6] estimated convergence using the estimated cart position data instead of the absolute estimation error of the acceleration. Impact 1 was not compared between the two real-time algorithms since both algorithms had different initial conditions. The study shows that the ensemble of LSTMs compares well with the MRAS-based algorithm.

Table 3: Average convergence time - dynamic cart tests.

	Impact 2	Impact 3	Impact 4
Ensemble of LSTMs (ms)	68	343	391
MRAS[6] (ms)	144	289	264

The use of the dominating attention layer weight showed promise in tracking the cart over time. Without the impact excitations, the weight associated with the 200 mm position dominates between 1285 and 3621 ms, while the weight associated with the 50 mm position dominates after 3769 ms. With the impact excitations, weights converge only post-impacts, with the weight associated with the 200 mm position dominating between 1271 and 4137 ms and 50 mm dominating after 6540 ms, and with the 100 mm weight slightly higher than others over 4402 and 6356 ms. It also appears that using a weighted average of positions when no weight is clearly dominating, or when weights are being adapted, may yield a better estimation of the cart position. This can be observed under no impact excitations when the cart moves from 200 to 50 mm, where the 200 mm weight ramps



(a) (b)

Figure 6: Results under dynamic cart location, showing the acceleration time series (top), estimation error (middle), and evolution of attention layer weights (bottom) for the system (a) without impact excitations and (b) with impact excitation.

down while the 50 mm weight ramps up. This is also shown during the movement from 50 mm to 200 mm, yet far less clearly given that weights had not yet initially converged. This relationship at the 50-200 mm transition is more clearly observable under the impact excitations, because the first impact enabled the initial convergence of weights. On the other and, the 200-50 mm transition is more overlooked by the weights.

Overall, the results support the hypothesis that the attention weights can be used to estimate the cart position even before the system converges. However, the main goal for this real-time algorithm is to predict high-rate dynamics, and a quick convergence time is desired. While further development of the algorithm could alleviate this shortcoming, the ensemble of LSTM cells could be used to conduct state estimation pre- and post-impact of a given dynamic system, for example, the state of an aircraft subjected to ballistic impacts. Another important remark is that the ensemble-of-LSTMs algorithm was designed for highly non-stationary systems, as discussed in.[7] The purpose of the study was to evaluate, on a simpler system, if the architecture of the ensemble of LSTMs could be leveraged for system identification. Otherwise, there exist better machine learning techniques that could be used for the purpose of estimating the cart position. For example, a more traditional RNN architecture could have been trained to directly output the estimated cart location.

5. CONCLUSION

This paper presented a real-time learning approach to high-rate state estimation. The approach consisted of an ensemble of LSTM cells trained using limited physical knowledge and joined by an attention layer used to predict time series. State estimation was conducted by investigating the evolution of the weights on the algorithm's attention layer. The performance of the algorithm was evaluated using experimental data acquired from DROPBEAR, a dedicated testbed for HRSHM research. Here, the state of interest was the position of a cart. Prior physical knowledge consisted of four different fundamental frequencies arising from four possible cart positions. Thus, four different LSTM cells were trained in the ensemble, each on a simple damped harmonic time series of appropriate frequency. Subsequently, the algorithm was adapted in real-time to conduct one-step ahead prediction of the measured acceleration time series. This was done on two sets of experiments. The first set used fixed cart locations, while the second set moved the cart back-and-forth along the beam.

Results on the fixed cart experiments showed that the dominating LSTM weight in the ensemble corresponded with the correct cart location. In the case of the dynamic cart, the correct cart location was observable post-convergence of the ensemble of LSTMs, which occurred quickly post impacts in the range of tens of milliseconds. A study of the evolution of attention weights showed that it was difficult to estimate the cart location while

moving. Convergence times compared well with those reported in the literature, and showed faster in the case of the dynamic cart after each impact.

Future work includes further developments of the algorithm to improve state estimation capabilities through the attention weights, in particular, while the cart is moving and the assessment of the algorithm on dynamics of higher complexity.

ACKNOWLEDGMENTS

The work presented in this paper is funded by the National Science Foundation under award number CISE-1937460. Their support is gratefully acknowledged. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the sponsor. The authors also acknowledge Dr. Janet Wolfson and Dr. Jonathan Hong for providing the experimental data.

REFERENCES

[1] Hong, J., Laflamme, S., and Dodson, J., "Study of input space for state estimation of high-rate dynamics," *Structural Control and Health Monitoring* **25**, e2159 (apr 2018).

[2] Hong, J., Laflamme, S., Dodson, J., and Joyce, B., "Introduction to state estimation of high-rate system dynamics," *Sensors* **18**, 217 (jan 2018).

[3] Dodson, J. C., Inman, D. J., and Foley, J. R., "Microsecond structural health monitoring in impact loaded structures," in [*Health Monitoring of Structural and Biological Systems 2009*], Kundu, T., ed., SPIE (mar 2009).

[4] Joyce, B., Dodson, J., Hong, J., and Laflamme, S., "Practical considerations for sliding mode observers for high-rate structural health monitoring," in [*Volume 2: Mechanics and Behavior of Active Materials; Structural Health Monitoring; Bioinspired Smart Materials and Systems; Energy Harvesting; Emerging Technologies*], American Society of Mechanical Engineers (sep 2018).

[5] Downey, A., Hong, J., Dodson, J., Carroll, M., and Scheppegrell, J., "Millisecond model updating for structures experiencing unmodeled high-rate dynamic events," *Mechanical Systems and Signal Processing* **138**, 106551 (apr 2020).

[6] Yan, J., Laflamme, S., Singh, P., Sadhu, A., and Dodson, J., "A comparison of time-frequency methods for real-time application to high-rate dynamic systems," *Vibration* **3**, 204–216 (aug 2020).

[7] Barzegar, V., Laflamme, S., Hu, C., and Dodson, J., "Multi-time resolution ensemble lstms for enhanced feature extraction in high-rate time series," *Sensors* (2021).

[8] Liu, X., Wang, Y., and Verriest, E. I., "Simultaneous input-state estimation with direct feedthrough based on a unifying MMSE framework with experimental validation," *Mechanical Systems and Signal Processing* **147**, 107083 (jan 2021).

[9] Dehghanpour, K., Wang, Z., Wang, J., Yuan, Y., and Bu, F., "A survey on state estimation techniques and challenges in smart distribution systems," *IEEE Transactions on Smart Grid* **10**, 2312–2322 (mar 2019).

[10] Xing, Y. and Lv, C., "Dynamic state estimation for the advanced brake system of electric vehicles by using deep recurrent neural networks," *IEEE Transactions on Industrial Electronics* **67**, 9536–9547 (nov 2020).

[11] Zhang, L., Wang, G., and Giannakis, G. B., "Real-time power system state estimation and forecasting via deep unrolled neural networks," *IEEE Transactions on Signal Processing* **67**, 4069–4077 (aug 2019).

[12] Belghazi, M. I., Baratin, A., Rajeswar, S., Ozair, S., Bengio, Y., Courville, A., and Hjelm, R. D., "Mine: Mutual information neural estimation," *ICML 2018* (Jan. 2018).